

Alice gives Bob a board composed of  $1 \times 1$  wooden squares and asks him to find the minimum cost of breaking the board back down into its individual squares. To break the board down, Bob must make cuts along its horizontal and vertical lines.

To reduce the board to squares, Bob makes horizontal and vertical cuts across the entire board. Each cut has a given cost,  $cost\_y[i]$  or  $cost\_x[j]$  for each cut along a row or column across one board, so the cost of a cut must be multiplied by the number of segments it crosses. The cost of cutting the whole board down into  $1 \times 1$  squares is the sum of the costs of each successive cut.

Can you help Bob find the minimum cost? The number may be large, so print the value modulo  $10^9 + 7$ .

For example, you start with a  $2 \times 2$  board. There are two cuts to be made at a cost of  $cost\_y[1] = 3$  for the horizontal and  $cost\_x[1] = 1$  for the vertical. Your first cut is across 1 piece, the whole board. You choose to make the horizontal cut between rows 1 and 2 for a cost of  $1 \times 3 = 3$ . The second cuts are vertical through the two smaller boards created in step 1 between columns 1 and 2. Their cost is  $2 \times 1 = 2$ . The total cost is  $3 + 2 = 5$  and  $5\%(10^9 + 7) = 5$ .

## Function Description

Complete the *boardCutting* function in the editor below. It should return an integer.

*boardCutting* has the following parameter(s):

- $cost\_x$ : an array of integers, the costs of vertical cuts
- $cost\_y$ : an array of integers, the costs of horizontal cuts

## Input Format

The first line contains an integer  $q$ , the number of queries.

The following  $q$  sets of lines are as follows:

- The first line has two positive space-separated integers  $m$  and  $n$ , the number of rows and columns in the board.
- The second line contains  $m - 1$  space-separated integers  $cost\_y[i]$ , the cost of a horizontal cut between rows  $[i]$  and  $[i + 1]$  of one board.
- The third line contains  $n - 1$  space-separated integers  $cost\_x[j]$ , the cost of a vertical cut between columns  $[j]$  and  $[j + 1]$  of one board.

## Constraints

- $1 \leq q \leq 20$
- $2 \leq m, n \leq 1000000$
- $0 \leq cost\_y[i], cost\_x[j] \leq 10^9$

## Output Format

For each of the  $q$  queries, find the minimum cost ( $minCost$ ) of cutting the board into  $1 \times 1$  squares and print the value of  $minCost \% (10^9 + 7)$ .

### Sample Input 0

```
1
2 2
2
1
```

### Sample Output 0

```
4
```

### Explanation 0

We have a  $2 \times 2$  board, with cut costs  $cost_y[1] = 2$  and  $cost_x[1] = 1$ . Our first cut is horizontal between  $y[1]$  and  $y[2]$ , because that is the line with the highest cost (2). Our second cut is vertical, at  $x[1]$ . Our first cut has a  $totalCost$  of 2 because we are making a cut with cost  $cost_y[1] = 2$  across 1 segment, the uncut board. The second cut also has a  $totalCost$  of 2 but we are making a cut of cost  $cost_x[1] = 1$  across 2 segments. Our answer is  $minCost = ((2 \times 1) + (1 \times 2)) \% (10^9 + 7) = 4$ .

### Sample Input 1

```
1
6 4
2 1 3 1 4
4 1 2
```

### Sample Output 1

```
42
```

### Explanation 1

Our sequence of cuts is:  $y[5]$ ,  $x[1]$ ,  $y[3]$ ,  $y[1]$ ,  $x[3]$ ,  $y[2]$ ,  $y[4]$  and  $x[2]$ .

Cut 1: Horizontal with cost  $cost_y[5] = 4$  across 1 segment.  $totalCost = 4 \times 1 = 4$ .

Cut 2: Vertical with cost  $cost_x[1] = 4$  across 2 segments.  $totalCost = 4 \times 2 = 8$ .

Cut 3: Horizontal with cost  $cost_y[3] = 3$  across 2 segments.  $totalCost = 3 \times 2 = 6$ .

Cut 4: Horizontal with cost  $cost_y[1] = 2$  across 2 segments.  $totalCost = 2 \times 2 = 4$ .

Cut 5: Vertical with cost  $cost_x[3] = 2$  across 4 segments.  $totalCost = 2 \times 4 = 8$ .

Cut 6: Horizontal with cost  $cost_y[2] = 1$  across 3 segments.  $totalCost = 1 \times 3 = 3$ .

Cut 7: Horizontal with cost  $cost_y[4] = 1$  across 3 segments.  $totalCost = 1 \times 3 = 3$ .

Cut 8: Vertical with cost  $cost_x[2] = 1$  across 6 segments.  $totalCost = 1 \times 6 = 6$ .

$totalCost = 4 + 8 + 6 + 4 + 8 + 3 + 3 + 6 = 42$ . We then print the value of  $42 \% (10^9 + 7)$ .